

SQLite数据库

主讲：万永权



学习 目标

AIMS

- 01 了解SQLite数据库的特点和体系结构
- 02 掌握SQLite数据库的建立和操作方法

8.3

数据库存储

8.3.1 SQLite数据库

- ✓ SQLite是一个2000年由D.Richard Hipp发布的开源嵌入式关系数据库
- ✓ 普通数据库的管理系统比较庞大和复杂，会占用了较多的系统资源
- ✓ 轻量级数据库SQLite的特点
 - ◆ 比传统数据库更适合用于嵌入式系统
 - ◆ 占用资源少，运行高效可靠，可移植性强
 - ◆ 提供了零配置（zero-configuration）运行模式

8.3

数据库存储

8.3.1 SQLite数据库

✓ SQLite数据库的优势

- ◆ 可以嵌入到使用它的应用程序中
 - 提高了运行效率
 - 屏蔽了数据库使用和管理复杂性
- ◆ 客户端和服务端在同一进程空间运行
 - 完全不需要进行网络配置和管理
 - 减少了网络调用所造成的额外开销
- ◆ 简化了数据库的管理过程
 - 应用程序更加易于部署和使用
 - 只需要把SQLite数据库正确编译到应用程序中

SQLite数据库

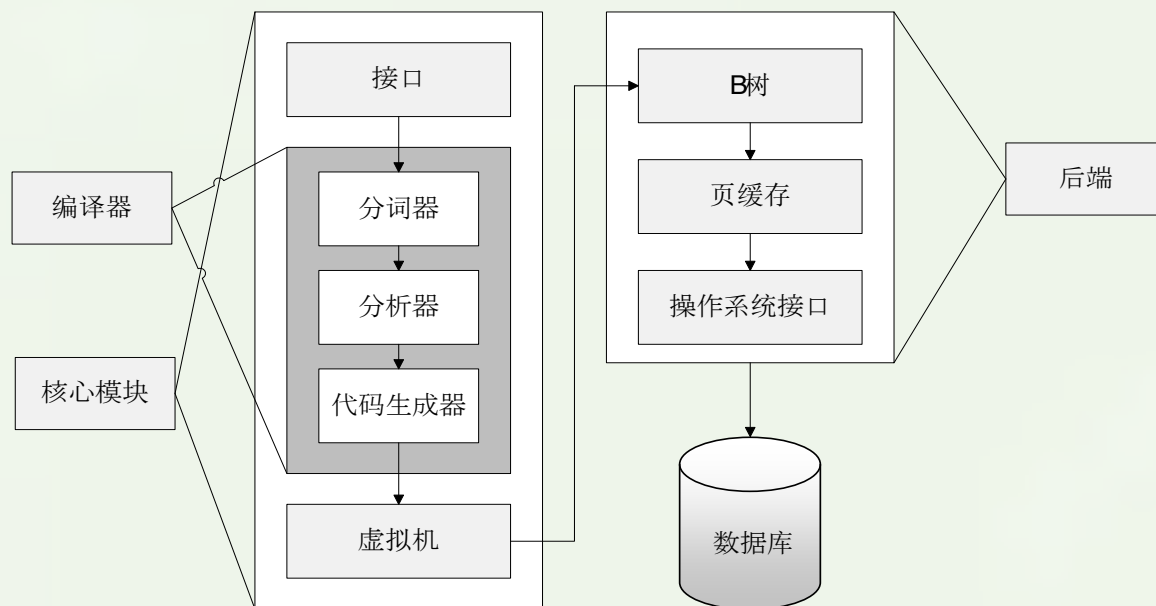
- ❖ SQLite是一种轻量级数据库系统，以嵌入式操作系统为设计目标，占用资源低，因此选择其作为手机操作系统的本地数据库，此外它还是开源的，任何人都可以使用。
- ❖ 许多开源项目（Mozilla，PHP，Python）都使用了SQLite，同样Android运行的环境包含了完整的SQLite。
- SQLite基本上**符合**SQL-92标准;
- 最大的不同就是对数据类型的支持(**5种数据类型**);
- 此外SQLite也**不支持**一些标准的SQL功能，特别是**外键约束**（FOREIGNKEY constrains），嵌套transcation和RIGHT OUTER JOIN和FULL OUTER JOIN，还有一些ALTER TABLE功能。
- 除了上述功能外，SQLite是一个完整的SQL系统，拥有完整的触发器，事务等等。

8.3

数据库存储

8.3.1 SQLite数据库

- ✓ SQLite数据库采用了模块化设计，模块将复杂的查询过程分解为细小的工作进行处理
- ✓ SQLite数据库由8个独立的模块构成，这些独立模块又构成了三个主要的子系统



8.3

数据库存储

8.3.1 SQLite数据库

✓ 接口

- ◆ 由SQLite C API组成，因此无论是应用程序、脚本，还是库文件，最终都是通过接口与SQLite交互

✓ 编译器

- ◆ 在编译器中，分词器和分析器对SQL语句进行语法检查，然后把SQL语句转化为便于底层处理的分层数据结构，这种分层的数据结构称为“语法树”
- ◆ 然后把语法树传给代码生成器进行处理，生成一种用于SQLite的汇编代码，最后由虚拟机执行

8.3

数据库存储

8.3.1 SQLite数据库

✓虚拟机

- ◆ SQLite数据库体系结构中最核心的部分是虚拟机，也称为虚拟数据库引擎 (Virtual Database Engine, VDBE)
- ◆ 与Java虚拟机相似，虚拟数据库引擎用来解释并执行字节代码
- ◆ 虚拟数据库引擎的字节代码由128个操作码构成，这些操作码主要用以对数据库进行操作，每一条指令都可以完成特定的数据库操作，或以特定的方式处理栈的内容

8.3

数据库存储

8.3.1 SQLite数据库

✓ 后端

- ◆ 后端由B-树、页缓存和操作系统接口构成，B-树和页缓存共同对数据进行管理
- ◆ B-树的主要功能就是索引，它维护着各个页面之间复杂的关系，便于快速找到所需数据
- ◆ 页缓存的主要作用是通过操作系统接口在B-树和磁盘之间传递页面

8.3

数据库存储

8.3.1 SQLite数据库

✓ 移植性

- ◆ 可以运行在Windows、Linux、BSD、Mac OS和一些商用Unix系统，比如Sun的Solaris或IBM的AIX
 - ◆ 嵌入式操作系统下，比如QNX、VxWorks、Palm OS、Symbian和Windows CE
- ✓ SQLite的核心大约有3万行标准C代码，因为模块化的设计使这些代码非常易于理解

8.3

数据库存储

8.3.2 手动建库

- ✓ 在Android系统中，每个应用程序的SQLite数据库被保存在各自的 /data/data/<package name>/databases 目录下
- ✓ 缺省情况下，所有数据库都是私有的，仅允许创建数据库的应用程序访问，如果需要共享数据库则可以使用ContentProvider
- ✓ 虽然应用程序完全可以在代码中动态的建立SQLite数据库，但使用命令行手工建立和管理数据库仍然是非常重要的内容，对于调试使用数据库的应用程序非常有用

8.3

数据库存储

8.3.2 手动建库

- ✓ 手动建立数据库指的是使用sqlite3工具，通过手工输入命令行完成数据库的建立过程
- ✓ sqlite3是SQLite数据库自带的一个基于命令行的SQL命令执行工具，并可以显示命令执行结果
- ✓ Android SDK的tools目录有sqlite3工具，同时，该工具也被集成在Android系统中
- ✓ 下面的内容将介绍如何连接到模拟器中的Linux系统，并在Linux系统中启动sqlite3工具，在Android程序目录中建立数据库和数据表，并使用命令在数据表中添加、修改和删除数据

8.3

数据库存储

8.3.2 手动建库

- ✓ 使用adb shell命令连接到模拟器的Linux系统，在Linux命令提示符下输入sqlite3可启动sqlite3工具
- ✓ 启动sqlite3后会显示SQLite的版本信息，显示内容如下：

```
1  # sqlite3
2  SQLite version 3.6.22
3  Enter ".help" for instructions
4  Enter SQL statements terminated with a ";"
5  sqlite>
```

8.3

数据库存储

8.3.2 手动建库

- ✓ 在启动sqlite3工具后，提示符从“#”变为“sqlite>”，表示用户进入SQLite数据库交互模式，此时可以输入命令建立、删除或修改数据库的内容
- ✓ 正确退出sqlite3工具的方法是使用.exit命令

```
1  sqlite> .exit  
2  #
```

8.3

数据库存储

8.3.2 手动建库

- ✓原则上，每个应用程序的数据库都保存在各自的/data/data/<package name>/databases目录下
- ✓但如果使用手工方式建立数据库，则必须手工建立数据库目录，目前版本无需修改数据库目录的权限

```
1  # mkdir databases
2  # ls -l
3  drwxrwxrwx root    root          2011-09-19 15:43 databases
4  drwxr-xr-x system  system        2011-09-19 15:31 lib
5  #
```

8.3

数据库存储

8.3.2 手动建库

- ✓ 在SQLite数据库中，每个数据库保存在一个独立的文件中
- ✓ 使用“sqlite3+文件名”的方式打开数据库文件，如果指定的文件不存在，sqlite3工具则自动创建新文件
- ✓ 下面的代码将创建名为people的数据库，在文件系统中将产生一个名为people.db的数据库文件

```
6  # sqlite3 people.db
7  SQLite version 3.6.22
8  Enter “.help” for instructions
9  Enter SQL statements terminated with a “;”
10 sqlite>
```


8.3

数据库存储

8.3.2 手动建库

- ✓ 下面的代码在数据库中，构造了一个名为peopleinfo的表
- ✓ 使用create table命令，关系模式为peopleinfo (_id, name, age, height)
- ✓ 表包含四个属性，_id是整型主键；name表示姓名，字符型，not null表示属性值一定要填写，不可以为空值；age表示年龄，整数型；height表示身高，浮点型

```
1  sqlite> create table peopleinfo
2  ...> (_id integer primary key autoincrement,
3  ...> name text not null,
4  ...> age integer,
5  ...> height float);
6  sqlite>
```

8.3

数据库存储

□8.3.2 手动建库

- ✓ 为了确认数据表是否创建成功，可以使用.tables命令，显示当前数据库中的所有表
- ✓ 从下面的代码中可以观察到，当前的数据库中仅有一个名为peopleinfo的表

```
1  sqlite> .tables  
2  poepleinfo  
3  sqlite>
```

8.3

数据库存储

□8.3.2 手动建库

- ✓也可以使用.schema命令查看建立表时使用的SQL命令
- ✓如果当前数据库中包含多个表，则可以使用[.schema 表名]的形式，显示指定表的建立命令

```
1  sqlite>.schema
2  CREATE TABLE peopleinfo
3  (_id integer primary key autoincrement,
4  name text not null,
5  age integer,
6  height float);
7  sqlite>
```



数据库存储

8.3.2 手动建库

- ✓ 下一步是向peopleinfo表中添加数据，使用insert into ... values命令
- ✓ 在下面的代码成功运行后，数据库的peopleinfo表将有三条数据，内容如下表所示：

```
1 sqlite> insert into peopleinfo values(null,'Tom',21,1.81);  
2 sqlite> insert into peopleinfo values(null,'Jim',22,1.78);  
3 sqlite> insert into peopleinfo values(null,'Lily',19,1.68);
```

- ✓ 因为_id是自动增加的主键，因此在输入null后，SQLite数据库会自动填写该项的内容

_id	name	age	height
1	Tom	21	1.81
2	Jim	22	1.78
3	Lily	19	1.68

8.3

数据库存储

□8.3.2 手动建库

- ✓ 在数据添加完毕后，使用select命令，显示peopleinfo数据表中的所有数据信息，命令格式为[select 属性 from 表名]
- ✓ 下面的代码用来显示peopleinfo表的所有数据

```
8  select * from peopleinfo;  
9  1|Tom|21|1.81  
10 2|Jim|22|1.78  
11 3|Lily|19|1.68  
12 sqlite>
```

8.3

数据库存储

8.3.2 手动建库

- ✓ 上面的查询结果看起来不是很直观，使用表格方式显示更应更符合习惯，因此可以使用.mode命令更改结果输出格式
- ✓ .mode命令除了支持常见的column格式外，还支持csv格式、html格式、insert格式、line格式、list格式、tabs格式和tcl格式
- ✓ 下面使用column格式显示peopleinfo数据表中的数据信息

```
1  sqlite> .mode column
2  sqlite> select * from peopleinfo;
3  1      Tom      21      1.81
4  2      Jim      22      1.78
5  3      Lily     19      1.68
6  sqlite>
```

8.3

数据库存储

8.3.2 手动建库

- ✓ 更新数据可以使用update命令，命令格式为[update 表名 set 属性="新值" where 条件]
- ✓ 更新数据后，同样使用select命令显示数据，确定数据是否正确更新
- ✓ 下面的代码将Lily的身高更新为1.88

```
1  sqlite> update peopleinfo set height=1.88 where name="Lily";
2  sqlite> select * from peopleinfo;
3  select * from peopleinfo;
4  1      Tom      21      1.81
5  2      Jim      22      1.78
6  3      Lily     19      1.88
7  sqlite>
```

8.3

数据库存储

8.3.2 手动建库

- ✓ 删除数据可以使用delete命令
- ✓ 命令格式为[delete from 表名where 条件]
- ✓ 下面的代码将_id为3数据从表peopleinfo中删除

```
1  sqlite> delete from peopleinfo where _id=3;  
2  sqlite> select * from peopleinfo;  
3  select * from peopleinfo;  
4  1      Tom      21      1.81  
5  2      Jim      22      1.78  
6  sqlite>
```


8.3

数据库存储

8.3.2 手动建库

✓ sqlite3工具还支持很多命令，可以使用.help命令查询sqlite3的命令列表，也可以参考右表

编号	命令	说明
1	.bail ON OFF	遇到错误时停止，缺省为OFF
2	.databases	显示数据库名称和文件位置
3	.dump ?TABLE? ...	将数据库以SQL文本形式导出
4	.echo ON OFF	开启和关闭回显
5	.exit	退出
6	.explain ON OFF	开启或关闭适当输出模式，如果开启模式将更改为column，并自动设置宽度
7	.header(s) ON OFF	开启或关闭标题显示
8	.help	显示帮助信息
9	.import FILE TABLE	将数据从文件导入表
10	.indices TABLE	显示表中所的列名
11	.load FILE ?ENTRY?	导入扩展库

8.3

数据库存储

8.3.2 手动建库

编号	命令	说明
12	.mode MODE ?TABLE?	设置输入格式
13	.nullvalue STRING	打印时使用STRING代替NULL
14	.output FILENAME	将输入保存到文件
15	.output stdout	将输入显示在屏幕上
16	.prompt MAIN CONTINUE	替换标准提示符
17	.quit	退出
18	.read FILENAME	在文件中执行SQL语句
19	.schema ?TABLE?	显示表的创建语句
20	.separator STRING	更改输入和导入的分隔符
21	.show	显示当前设置变量值
22	.tables ?PATTERN?	显示符合匹配模式的表名
23	.timeout MS	尝试打开被锁定的表MS毫秒
24	.timer ON OFF	开启或关闭CPU计时器
25	.width NUM NUM ...	设置"column"模式的宽度